

International Journal of Concrete Structures and Materials
Vol.9, No.2, pp.159–172, June 2015
DOI 10.1007/s40069-015-0097-4
ISSN 1976-0485 / eISSN 2234-1315



Residual Strength of Corroded Reinforced Concrete Beams Using an Adaptive Model Based on ANN

Ashhad Imam^{1),*}, Fatai Anifowose²⁾, and Abul Kalam Azad³⁾

(Received June 12, 2014, Accepted January 28, 2015, Published online March 7, 2015)

Abstract: Estimation of the residual strength of corroded reinforced concrete beams has been studied from experimental and theoretical perspectives. The former is arduous as it involves casting beams of various sizes, which are then subjected to various degrees of corrosion damage. The latter are static; hence cannot be generalized as new coefficients need to be re-generated for new cases. This calls for dynamic models that are adaptive to new cases and offer efficient generalization capability. Computational intelligence techniques have been applied in Construction Engineering modeling problems. However, these techniques have not been adequately applied to the problem addressed in this paper. This study extends the empirical model proposed by Azad et al. (Mag Concr Res 62(6):405–414, 2010), which considered all the adverse effects of corrosion on steel. We proposed four artificial neural networks (ANN) models to predict the residual flexural strength of corroded RC beams using the same data from Azad et al. (2010). We employed two modes of prediction: through the correction factor (C_f) and through the residual strength (M_{res}). For each mode, we studied the effect of fixed and random data stratification on the performance of the models. The results of the ANN models were found to be in good agreement with experimental values. When compared with the results of Azad et al. (2010), the ANN model with randomized data stratification gave a C_f -based prediction with up to 49 % improvement in correlation coefficient and 92 % error reduction. This confirms the reliability of ANN over the empirical models.

Keywords: corrosion, reinforced concrete beam, flexural strength, artificial neural networks.

1. Introduction

Corrosion of reinforcement steel has been proved to be a major cause of deterioration of reinforced concrete (RC) structures, resulting in the reduction of the service life of concrete structures. A substantial amount of research related to reinforcement corrosion has been carried out in the past, addressing various issues related to the corrosion process, its initiation and damaging effects. Assessment of the flexural strength of corrosion-damaged RC members has been studied (Azad et al. 2010; Cabrera 1996; Huang and Yang 1997; Rodriguez et al. 1997; Uomoto and Misra 1988). A number of studies have also been conducted on the prediction of residual flexural strength of corroding concrete beams (Azad et al. 2007; Mangat and Elgarf 1999; Nokhasteh and Eyre

1992; Ravindrarajah and Ong 1987; Tachibana et al. 1990; Wang and Liu 2008; Jin and Zhao 2001). Some of these studies had been conducted in the laboratory. They involve the casting of concrete beam specimens sometimes in large scale, in the order of meters in dimension (Ou et al. 2012), and sometimes in small scale, in the order of millimeters (Azad et al. 2007; Mangat and Elgarf 1999; Nokhasteh and Eyre 1992; Revathy et al. 2009; Tachibana et al. 1990; Wang and Liu 2008; Jin and Zhao 2001). The specimens are then subjected to various degrees of corrosion damage after which the samples are tested for their bending or flexural performances. These procedures take a lot of time as some of the specimens need to be left for several days to attain their required degree of corrosion. They also require the use of expensive and specialized laboratory equipment, exorbitant man-hours and concerted effort. An average experiment can take up to 6 months to complete. Though, experiments are the best sources of real data but the associated costs often make them prohibitive.

In order to reduce the completion time and avoid the cost associated with such studies without compromising on accuracy, some attempts have been made on the use of numerical modeling methods (Azad et al. 2007, 2010; Cabrera 1996; Coronelli and Gambarova 2004; Ou et al. 2012). These methods are however static and cannot be generalized well on datasets outside those for which they were designed. Most of them do not consider the non-linearity of the attributes of the natural phenomena involved in the corrosion

¹⁾Department of Civil Engineering, King Fahd University of Petroleum & Minerals, Dhahran 31261, Saudi Arabia.

*Corresponding Author; E-mail: ashhad.ce@gmail.com

²⁾Center for Petroleum and Minerals, Research Institute, King Fahd University of Petroleum & Minerals, Dhahran 31261, Saudi Arabia.

³⁾Department of Civil & Environmental Engineering, College of Engineering Sciences, King Fahd University of Petroleum & Minerals, Dhahran 31261, Saudi Arabia.

Copyright © The Author(s) 2015. This article is published with open access at Springerlink.com

process. Since corrosion is a natural process, it is expected that its attributes be non-linearly related to the corrosion property being studied. Hence, modeling the process with linear relations is inadequate. To make such models more generalized, they need to be recalibrated with new sets of data. Doing this will result in re-generating new sets of coefficients to evolve a new model, which requires considerable time and effort.

With the limitations in the experimental and theoretical methods, the quest for cost-effective, easy to use and adaptive models that offer scalability and efficient generalization capability to new cases continues. With the huge amount of data generated from various experiments over the years, robust data mining techniques that are based on computational intelligence (CI) and machine learning paradigms are hypothesized to be capable of overcoming the limitations of the conventional methods. The prediction and generalization capability of artificial neural networks (ANN) had been investigated in this paper. With the capability of ANN to handle the non-linearity in natural phenomena such as corrosion, coupled with its capability to adaptively learn from hidden patterns in experimental data, we presented two optimized models of ANN to efficiently predict the flexural strength of corrosion-damaged RC beams.

The motivations for choosing the proposed ANN models are:

- ANN is the most commonly used of the CI techniques in various application areas (Abdalla et al. 2007; VanLuchene and Sun 1990; Waszczyszyn and Ziemiański 2001; Wu et al. 1992).
- Though ANN has been applied in modeling other civil engineering problems, they have not been adequately applied to the problem of estimating the residual flexural strength of corrosion-damaged RC beams, which is the focus of this paper.
- ANN is easy to use and understand by researchers outside the Computer Science field.
- Following the principle of Occam's Razor (Jefferys and Berger 1991), starting an investigation with a simple model like ANN is preferred to using more complex and state-of-the-art techniques.
- Since the performance of a model is determined by the nature of the problem (represented by data), there is no guarantee that using a more sophisticated algorithm will perform better. This agrees with the No Free Lunch theorem (Wolpert and Macready 1997).

Further to ensuring simplicity in design and implementation, proposing optimization- and feature selection-based hybrid models is considered undesirable. Optimization algorithms such as Particle Swarm, Genetic Algorithm, Bee Colony, Ant Colony, etc. are based on heuristic and exhaustive search paradigms (Bies et al. 2006). Due to this, they take much time to converge, require considerable memory resources and are complex. Also, since this study does not involve high-dimensional dataset, proposing feature selection-based hybrid models will be of no use.

The rest of this paper is organized as follows: Sect. 2 presents a literature survey on the proposed study. Section 3

gives a brief background on the proposed technique as well as the previously published empirical equation. Section 4 describes the datasets used for this study and the details of the proposed methodology. Results are presented and discussed in Sect. 5 while conclusions, highlighting the contributions of this study as well as its limitations, are presented in Sect. 6.

2. Literature Survey

The estimation of the flexural strength of corroded RC beams has been a focus of keen research for almost three decades (Cabrera 1996; Huang and Yang 1997; Ravindrarajah and Ong 1987; Rodriguez et al. 1997; Uomoto and Misra 1988). That shows the importance of this phenomenon in the construction engineering field. One of the earliest studies on this subject include the experiment carried out by Ravindrarajah and Ong (1987) to study the effect of corrosion on steel bars in mortar with the use of an accelerated corrosion technique. This was followed by Uomoto and Misra (1988) who studied the behavior of concrete beams and the changes in columns as corrosion of reinforcing bars increases. They presented an idea on when to repair the structures in marine environment. Another interesting research was carried out by Rodriguez et al. (1997) who induced corrosion to some RC beams. The data extracted from the experiment was used to develop some numerical models for the assessment of concrete structures affected by steel corrosion and other deterioration mechanisms. A similar work was carried out by Huang and Yang (1997) who tested thirty-two concrete beams in order to assess their structural behavior due to corrosion. The afore-mentioned studies were based on experimental methods whose limitations have been highlighted in Sect. 1.

Numerical methods of estimating the effect of corrosion have been applied since the experimental studies started. One of the earliest of such studies is that of Cabrera (1996) who used laboratory data to derive numerical models to relate the rate of corrosion to cracking and loss of bond strength. Rodriguez et al. (1997) used one of the Euro Code 2 conventional models to predict the ultimate bending moment and shear force. Later, Coronelli and Gambarova (2004) used a finite-element-based numerical procedure to estimate the bond deterioration index of concrete beams under corrosive conditions. Azad et al. (2007) and Al-Gohi (2008) employed a regression analysis method on the data obtained from experiments to predict the residual flexural strength of RC beams. They formulated a correction factor that can be used to calculate the flexural strength of a corroded beam using the reduced area of corroded bars. They later improved this correction factor in Azad et al. (2010) using new sets of data. When compared with Azad et al. (2007), they concluded that the improved model yielded values that are in good agreement with the test data, lending confidence to the proposed method to serve as a reliable analytical tool to predict the flexural capacity of a corroded concrete beam. The most recent study is the empirical equation proposed by Ou et al. (2012) who concluded that the results closely corresponds to experimental values.

It is well known that all these regression analysis and analytical modeling techniques, though performing well in their respective applications, do not handle the non-linearity between independent variables and their target values. They are rather based on the assumption that the dependent variables are linearly related to one or more of the independent variables. They do not consider the hidden and the non-linear relationships that exist in such natural phenomena. CI techniques, on the other hand, have the capability to extract hidden patterns and “learn” from historical knowledge to make predictions about unknown future cases (Eskandari et al. 2004). Applications of CI techniques have demonstrated superior performance over regression analysis and analytical modeling techniques. ANN has featured in a number of engineering problems with excellent performance (Castillo et al. 2001; Mohaghegh 1995; Rafiq et al. 2001; Tsai and Hsu 2002).

A set of pragmatic guidelines for designing ANN for engineering applications were proposed by Rafiq et al. (2001). Hsu and Chung (2002) evolved a model of damage diagnosing for RC structures using the ANN technique. The network learning procedure showed that the rate and the accuracy of the convergence is acceptable while the test results showed that the technique is efficient for the problem. ANN has been used to predict the shear strength (Cabrera 1996), deformation capacity (Inel 2007) and shear resistance (Abdalla et al. 2007) of various shapes of RC beams. The prediction of residual flexural strength has not been addressed with CI techniques. We intend to fill this research gap by proposing ANN models for this problem.

The success of the few attempts at utilizing the learning and predictive power of ANN in construction engineering as discussed above is the major motivation for this study. The main objective of this study is to use the ANN technique in the prediction of residual flexural strength of corroded RC beams. This technique utilizes the capability of the supervised machine learning concept to model the complex non-linear relationship between the properties of RC beams and their independent variables.

The next section gives a brief theoretical background of ANN and the necessary details of the empirical model proposed by Azad et al. (2010).

3. Theoretical Background

3.1 Artificial Neural Networks

Artificial Neural Networks (ANN) was inspired by the functioning of the human brain. It is an emulation of the biological nervous system that is made up of several layers of neurons (nodes) interconnected by links. Each node is assigned a weight. ANN can be thought of as a “computational system” that accepts inputs and produces outputs (Baughman 1995). Figure 1 shows how the human nervous system is mapped to evolve a typical ANN structure. The dendrites and synapses that serve as receptacles of excitatory signals in Fig. 1a is equivalent to the input layers of ANN in Fig. 1b from where input variables are admitted into the system. As

the nucleus gathers all input signals and prepares them for processing (as shown in Fig. 1a), the input signals in ANN are multiplied by the weights and biases, and aggregated in the summation layer (in Fig. 1b) where they are “excited” with an activation function. Finally, while the gathered signals are sent through the axon (Fig. 1a) to the brain for processing, the optimized results produced by ANN are sent to the output layer for interpretation and decision-making processes.

The individual inputs: x_1, x_2, \dots, x_k are multiplied by weights and the weighted values are fed to the summing junction (\sum). Their sum is simply wx , which is the dot product of the matrix w and the vector x . The neuron has a bias μ_i , which is summed up with the weighted inputs to form the predicted output y_i subject to the transfer function f . The input vector enters the network through the weight matrix w expressed as:

$$w = \begin{bmatrix} w_{11}, w_{12}, \dots, w_{1k} \\ w_{21}, w_{22}, \dots, w_{2k} \\ \vdots \\ w_{i1}, w_{i2}, \dots, w_{ik} \end{bmatrix} \quad (1)$$

where i is the number of neurons and k is the number of inputs.

The ANN processes are mathematically represented and generalized as:

$$y_i = f \left(\sum_k w_{ik} x_k + \mu_i \right) \quad (2)$$

where x_k are inputs to the neuron, w_{ik} are weights attached to the inputs to the neuron, μ_i is a threshold, offset or bias, $f(\cdot)$ is a transfer function and y_i is the output of the neuron. The transfer function $f(\cdot)$ can be any of: linear, non-linear, piecewise linear, sigmoidal, tangent hyperbolic and polynomial functions.

As each input x_k is applied to the network, the network output is compared to the target. The error is calculated as the difference between the target output and the network output. The goal is to minimize the average of the sum of these errors. The least mean square (LMS) algorithm adjusts the weights and biases of the linear network so as to minimize this mean square error:

$$mse = \frac{1}{Q} \sum_{k=1}^Q e(k)^2 = \frac{1}{Q} \sum_{k=1}^Q (t(k) - a(k))^2 \quad (3)$$

where Q is the number of data samples, e is the error criterion, t is the original target values and a is the model predicted values.

The mean square error performance index for the linear network is a quadratic function. Thus, the performance index will either have one global minimum, a weak minimum, or no minimum, depending on the characteristics of the input vectors. Specifically, the characteristics of the input vectors determine whether or not a unique solution exists.

The number of neurons in the input layer corresponds to the number of input variables fed into the neural network.

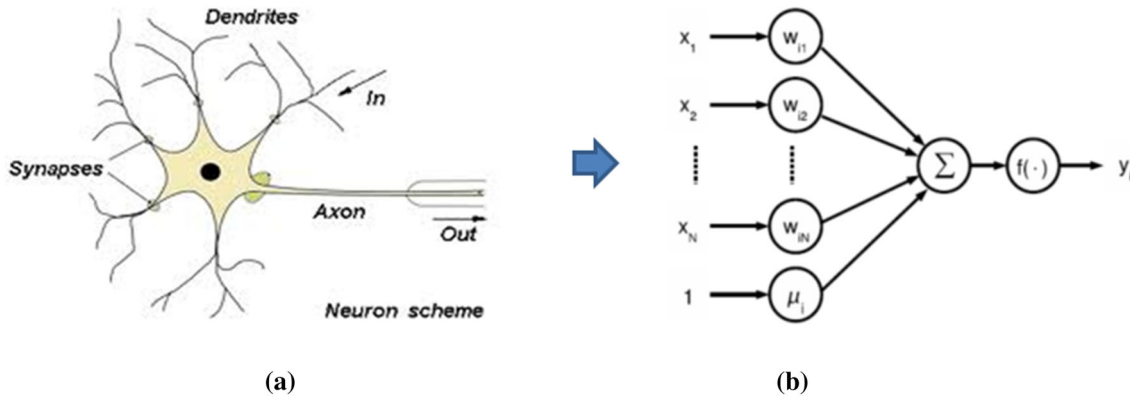


Fig. 1 The Architecture of artificial neural networks.

The number of hidden layer(s) and the corresponding number of neurons in each determine the “power” of the network. This is determined through the training process. The complexity of a problem determines the “power” it requires. Using more power than necessary will lead to the model being overfitted. In this case, the models perform well on the training data but poorly on the validation data (Hastie et al. 2009). On the other hand, using less power than required will lead to underfitting (Hastie et al. 2009). In this case, the model performs poorly on both training and validation datasets.

In order to concentrate more on the subject of this study, readers are referred to (Petrus et al. 1995) for more details of the architecture, structure and mathematical bases of ANN. This technique has caught the interest of most researchers and has today become an essential part of the technology industry, providing a good ground for solving many of the most difficult prediction problems in various areas of engineering applications (Baughman 1995; Guler 2005; Inan et al. 2006; Li and Jiao 2002; Moghadassi et al. 2009; Mohaghegh 1995; Nascimento et al. 2000; Phung and Bouzardoum 2007; Übeyli 2009). ANN has also gained vast popularity in solving various Civil Engineering problems (Baughman 1995; Beale and Demuth 2013; Chen et al. 1995; Flood and Kartam 1994; Hasancebi and Dumlupinar 2013; Kang and Yoon 1994; Kirkegaard and Rytter 1994; Neaupane and Adhikari 2006; Pandey and Barai 1995; Rafiq et al. 2001).

3.2 The Previously Published Empirical Model

Azad et al. (2010) proposed the following two-step procedure to predict the residual flexural strength of corroded beams for which the cross-sectional details, material strengths, corrosion activity index $I_{corr}T$, and diameter of rebar, D were known. The procedure used is:

- First, the moment capacity ($M_{th,c}$) was calculated using reduced cross-sectional area of tensile reinforcement, A'_s , in the conventional manner.
- The computed value of $M_{th,c}$ was then multiplied by a correction factor (C_f) to obtain the predicted residual flexural strength of the beam (M_{res}) using this relation:

$$M_{res} = C_f M_{th,c} \quad (4)$$

where C_f was assumed to represent the combined effect of bond loss and factors pertaining to loss of flexural strength other than the reduction of the metal area.

- The value of C_f was taken as a function of the two important variables, namely $I_{corr}T$ and D . Finally, using regression analysis of the data and a gravimetric analysis of the weight loss of steel (Beale and Demuth 2013), the following empirical equation for the correction factor C_f was derived:

$$C_f = \frac{5.0}{D^{0.54}(I_{corr}T)^{0.19}}; C_f \leq 1.0 \quad (5)$$

where D is the diameter of rebar in mm, I_{corr} is the corrosion current density in mA/cm² and T is the duration of corrosion in days.

The residual flexural strength of a rectangular beam could be determined using Eq. (4) by substituting the value of the calculated C_f from Eq. (5).

As stated earlier, the limitation of the above equations is that they were based on the assumption that $I_{corr}T$ and D are linearly related to the residual flexural strength. In such a natural phenomenon as corrosion, this will not give an optimal solution to the estimation problem. Basing construction projects on the results of this equation may result in sub-optimal life spans and increase maintenance costs. In order to overcome this limitation, ANN models, with their capability to utilize the non-linear relationship of the variables and their ability to extract hidden patterns from datasets, are proposed.

4. Research Methodology

4.1 Description of Data

For effective validation and comparison with the results of previous work, we designed and implemented our ANN models by using the same experimental data obtained and used by Azad et al. (2010) to predict the residual strength of corroding concrete beams. The data was obtained from an experiment consisting of 48 RC beams of different cross-sections and reinforcements. The beams were made of three different depths viz. 215, 265 and 315 mm, two different

diameters of tension bars viz. 16 and 18 mm, and different durations of corrosion. The corrosion was induced by applying a direct current at a constant rate of 1.78 mA/cm². Out of the 48 beams, 36 were subjected to accelerated corrosion. Both the corroded and un-corroded beams were tested in a four point bend to find their load carrying capacity using a span length of 900 mm and a flexure span of 200 mm. Statistically, the values of both C_f and M_{res} follow normal distribution as shown in the histograms in Figs. 2 and 3 respectively.

The results of the basic descriptive statistics (maximum, minimum, range, mean, variance, standard deviation, skewness, and kurtosis) that further describe the dataset are presented in Table 1. The maximum statistic is the largest or the greatest value in a set of data. The minimum is the smallest or the least value in a given set of data. The range, the arithmetical difference between the maximum and the minimum of a set of data, is a statistical measure of the spread of a dataset. The most common expression for the mean of a statistical distribution with a discrete random variable is the mathematical average of all the terms. The variance, a measure of dispersion in a data, is the average squared distance between the mean and each item in the population or sample.

The standard deviation, also a measure of dispersion, is the positive square root of the variance. An advantage of the standard deviation (compared to the variance) is that it expresses dispersion in the same units as the original values in the sample or population. Skewness is a measure of the extent to which a probability distribution of a real-valued random variable “leans” to either side of the mean. The skewness value can be positive or negative, or even undefined. When the skewness is positive, then it is said that the distribution is skewed to the right of the mean. Similarly, when it is negative, the distribution is skewed to the left. A zero skewness is said to be undefined. Kurtosis is a measure of the “peakedness” or the “flatness” of the probability distribution of a real-valued random variable. In a similar way to the concept of skewness, kurtosis is a descriptor of the shape of a probability distribution.

Table 1 shows that the predictor variables are diameter (D) and corrosion activity index ($I_{corr}T$) while the target properties are correction factor (C_f) and residual strength (M_{res}). The variable D has a range of 2 with the maximum and minimum values being 18 and 16 respectively and an average value of 17.03. The variable $I_{corr}T$ has a higher range of 30.16 with the maximum and minimum values being 32.06 and 1.90 respectively and an average value of 15.36. The maximum, minimum, range and average values of the C_f are much less than those of M_{res} . The values of C_f range between 0.5 and 1.08 (Table 1) with most of the values concentrated around 0.73 (Fig. 2) while M_{res} has its range between 16.1 and 66 (Table 1) and those with the highest frequency around 36 (Fig. 3). The variances and standard deviations of D and C_f are much less than those of $I_{corr}T$ and M_{res} respectively. The distribution of D is skewed to the left with a negative value while that of others is skewed to the right with positive values. The values of the skewness, together with the statistical trends shown in Figs. 2 and 3, are an indication that the data is not symmetrical. The negative kurtosis of all the predictor and target variables shows that the distribution of the data is flat rather than being peaked or Gaussian. All these are an indication that the data is typical of real-life experimental data and not simulated.

In order to simulate the practical application of the ANN models, the data was divided into training and testing subsets. The training subset represented the available experimental data comprising both the cross-sectional details and their corresponding flexural strength estimations while the testing subset represented only the experimental measurements without their equivalent flexural strength estimations. The models used the hidden non-linear relationship between the experimental measurements and the target values to train the models. The trained models were then used to predict the required flexural strength estimations given the desired experimental measurements. This was intended to save time, cost and man-hours while improving the accuracy.

To achieve the above implementation objective, the data was divided in the 70:30 ratio in which 70 % was used for training

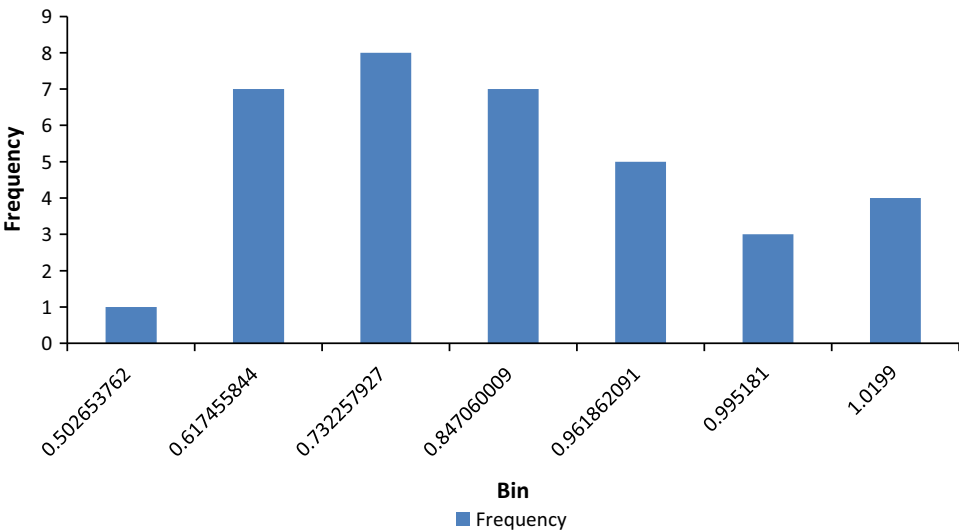


Fig. 2 The histogram of C_f values.

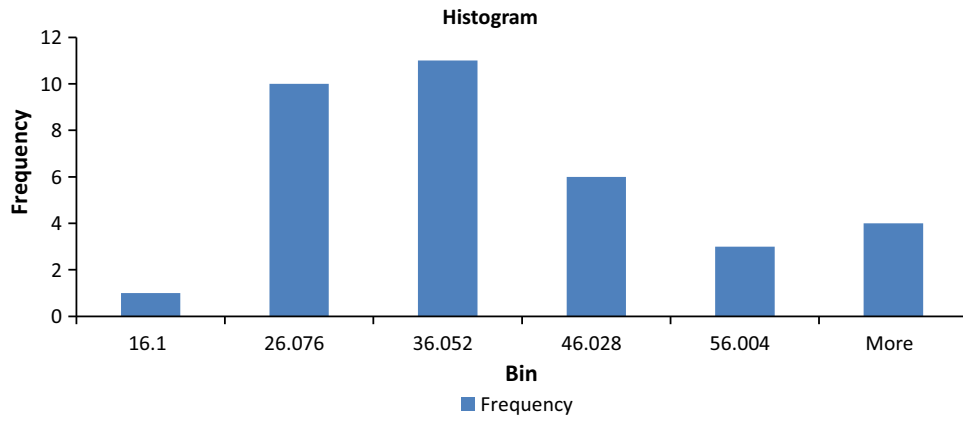


Fig. 3 The histogram of M_{res} values.

Table 1 Basic descriptive statistics of the data.

Statistics	Predictor variables		Target properties	
	Diameter (D)	Corrosion activity index (I_{corr} T)	Correction factor (C_f)	Residual strength (M_{res})
Maximum	18.00	32.06	1.0767	65.98
Minimum	16.00	1.90	0.5026	16.10
Mean	17.03	15.36	0.7749	34.60
Range	2.00	30.16	0.5740	49.88
Variance	0.99	76.38	0.0275	167.11
Standard deviation	1.01	8.87	0.1683	13.12
Skewness	-1.77	0.70	0.4308	0.40
Kurtosis	-2.12	-0.93	-1.2438	-0.17

and the remaining 30 % for testing. This is similar to the k-fold cross-validation technique and follows the standard machine learning paradigm that requires that the training subset should be much more than the testing. This is partly meant to avoid the overfitting and underfitting problems (Hastie et al. 2009). The authors experimented with two data stratification strategies: fixed and random. In the former, we used the first 70 % for training and the remaining for testing. However, in the latter, the authors took 70 % of the randomized sample of the data for training and the remaining for testing. The authors opined that the latter option is more representative of the experimental measurements and has the potential to avoid the bias that is usually associated with the choice of the training data. Hence, it was hypothesized that the results from the randomized stratification will be of more confidence. The outcome of this hypothesis is revealed in Sect. 5.

4.2 Criteria for Models Evaluation

For ease of comparison and due to their common use in predictive modeling literature, we used the comparative coefficient of determination (R^2) and root mean square error (RMSE) to evaluate the comparative performance of the ANN models and the empirical equation of Azad et al. (2010). The R^2 measures the statistical correlation between the predicted (y) and actual values (x). It is expressed as:

$$R^2 = \frac{n \sum xy - (\sum x)(\sum y)}{\sqrt{n(\sum x^2) - (\sum x)^2} \sqrt{n(\sum y^2) - (\sum y)^2}} \quad (6)$$

The RMSE is a measure of the spread of the actual x values around the average of the predicted y values. It computes the average of the squared differences between each predicted value and its corresponding actual value. It is expressed as:

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (x_i - y_i)^2}{n}} \quad (7)$$

4.3 Details of Design and Implementation of the ANN Models

Four different ANN models were developed in this study: A model each to directly predict the flexural strength of corroded RC beams (M_{res}) and indirectly through the prediction of the correlation factor (C_f). The implementation of each of these models was carried out with fixed and random data stratifications as discussed in Sect. 4.1. Each of the models is basically made up of a two-hidden-layer configuration with two neurons in the input layer and one in the output layer. Our choice of this configuration is based on the report of Beale and Demuth (2013) that a 2-layer neural network is enough to solve most problems. The two neurons in the input layer represent the diameter of reinforced steel

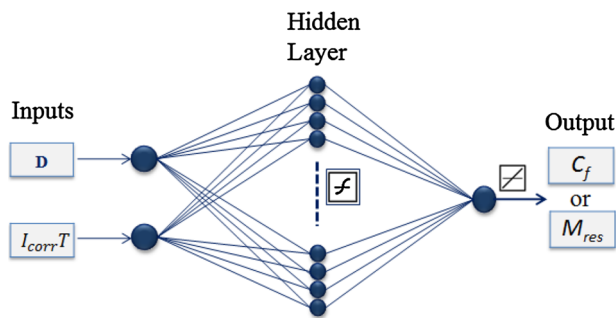


Fig. 4 Basic ANN architecture used in this work.

(D) and corrosion activity index ($I_{corr}T$) while the output neuron represents each of the target properties, C_f and M_{res} . Figure 4 is a sketch of the basic architecture used in this study showing the input layer, hidden layer, output layer and the transfer functions in the hidden and output layers.

Following the assumption that no problem would require an ANN model with more than 50 neurons in each hidden layer (Beale and Demuth 2013), the optimal number of hidden neurons was investigated from 1 through 50 while trying different learning algorithms and activation functions for the hidden and output layers. This procedure to search for optimal parameters is necessary since each problem requires different values of these parameters for optimal performance. Choosing the appropriate number of hidden neurons is essential for the ultimate performance of ANN models. If the number is too small, the model will not adequately capture the pattern that is hidden in the data. This leads to underfitting, which is characterized by a generally poor performance in both training and testing (Hastie et al. 2009). If the number is too large, the model will exert too much energy than necessary to solve the problem. This leads to overfitting characterized by a model performing excellently well in training but poor in generalization on new cases (Hastie et al. 2009).

In order to avoid cases of underfitting and overfitting, we followed the standard training procedure of ANN as presented in the following algorithm:

1. Divide data into training and testing subsets.
2. Initialize the weights and biases to some random initial values.
3. Find the dot product of weights and input vector.
4. Feed forward the results of each hidden layer to the output layer.
5. Compare the network outputs with the original target values.
6. If the error is equal or less than the pre-set error goal then
 7. Use the network to predict the target for the testing subset.
 8. Present prediction results.
 9. Go to 13.
10. Otherwise
 11. Back-propagate results to the end of the input layer.
 12. Go to 2.
13. End.

While the optimal number of hidden neurons obtained for each model of ANN is presented under the description of the respective models in the following sections, the other parameters used in the design of the ANN models are:

- Number of training epochs = 100
- Error goal = 0.001
- Training algorithm = Levenberg–Marquardt
- Error criterion = Mean squared error
- Transfer function in the hidden layer = Sigmoidal
- Transfer function in the output layer = Purelin

The first two parameters above were used to determine when to stop the training process: either the attainment of the error goal or using the most optimal configuration obtained on reaching the maximum number of epochs. Following the standard machine learning paradigm, the feed-forward back-propagation ANN models were trained by feeding a matrix of training data complete with the input and target values. The main objective of the training process is to optimize the connection weights that contribute toward reducing the prediction errors between the predicted and actual target values to a satisfactory level (the preset error goal). This process is carried out through the minimization of the defined error function by updating the connection weights.

The feed-forward back-propagation algorithm operates in two passes: the feed-forward pass and the back-propagation pass. During the feed-forward pass, the initial weights (either fixed or randomized) are applied on the input matrix and propagated through the hidden layer(s) to the output layer. At the output layer, the model is evaluated with a part of the training data (called validation data) and the error between the prediction results and the actual values of the target variable are compared with the error goal. If the error obtained is higher than the goal, the back-propagation pass is launched. This is where the connection weights are propagated back to the input layer where a new set of weights are computed (or re-initialized) and the feed-forward process continues. This loop continues until either the error goal is achieved or the pre-defined number of epochs is attained. The optimized parameters obtained at this point were then used. After the errors are minimized, the trained models with all the optimized parameters can be saved as virtual models and used for the prediction of the target values given any set of hypothetical or real-life input values to be experimented with.

The following sections explain the optimization details of each of the four models.

4.3.1 Prediction of C_f with Fixed Data Stratification

In this model (called Model 1-1), the data was divided by taking the first 70 % and the remainder of the samples for training and testing respectively. The model was used to predict C_f , which would be multiplied with the computed moment capacity ($M_{th,c}$) using Eq. (2). The result of the search for the optimal number of neurons in the hidden layer is shown in Fig. 5. From the plot, the optimal number of hidden neurons for this model is 13. This corresponds to the minimum number (x-axis) that gave the highest testing accuracy (y-axis) with the least overfitting. The least overfitting is determined by the least separation between the training and testing points. Although, several points qualify for this criteria but the most optimal was chosen.

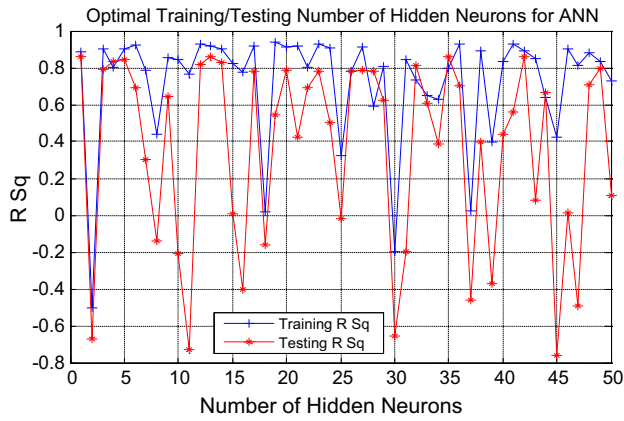


Fig. 5 Optimal number of hidden neurons for Model 1-1.

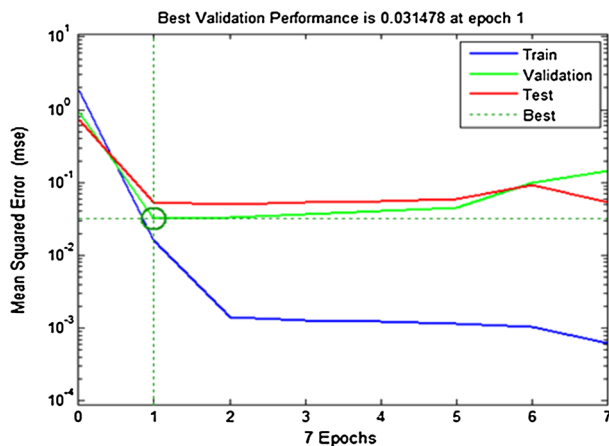


Fig. 6 Results of the training and validation for Model 1-1.

Figure 6 shows how the minimum error (hence the optimal model) was attained with respect to the validation and testing during the training process. Out of the 100 pre-defined for the model, the minimum error was attained within 7 epochs after which the validation error continued to increase. Hence, the best validation error attained so far at the first epoch was chosen. It could be seen that the errors decreased sharply at the beginning. However, after the first epoch, while training error continues to decrease, the test and validation errors could not converge in the same manner. Hence, the model decided to stop the training process and selected the best validation error attained so far.

4.3.2 Prediction of C_f with Random Data Stratification

This model (called Model 1-2) uses the random stratification of the data. A randomly selected 70 % and the remainder of the samples were taken for training and testing respectively. The randomization process ensured that each sample has equal chance of being selected for training or testing. This also ensures that there is a good mix of the data and all experimental cases are represented in each subset. Unlike the case of Model 1-1 (explained in Sect. 4.3.1), the randomization procedure avoids bias and skewness in the data given to the model for training and testing.

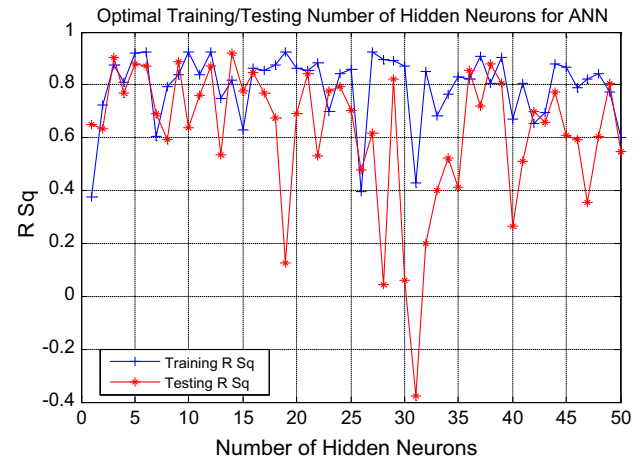


Fig. 7 Optimal number of hidden neurons for Model 1-2.

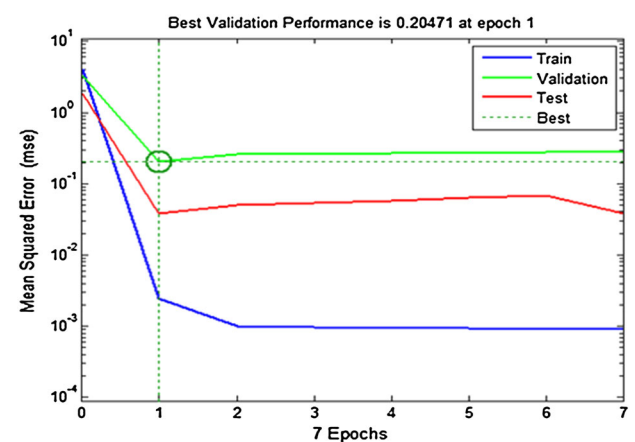


Fig. 8 Results of the training and validation for Model 1-2.

The optimal number of hidden neurons found and used for this model was 3 (as shown in Fig. 7). As defined in Sect. 4.2.1, this corresponds to the minimum number (x-axis) that gave the highest testing accuracy (y-axis) with the least overfitting. Similar to Fig. 4, the attainment of the optimal validation error is shown in Fig. 8. Seeing that the validation error could not converge after 7 epochs, the learning algorithm stopped the training process. Hence, the training process that gave the best error attained so far at epoch 1 was chosen.

4.3.3 Prediction of M_{res} with Fixed Data Stratification

Unlike the previous two, this model (named model 2-1) was developed to directly predict the residual flexural strength (M_{res}) without using C_f but with the same set of input variables. Like in model 1-1, the first 70 % of the data were used for training an ANN model while the last 30 % was used for testing and validation. The optimal number of hidden neurons found to be optimal for this model was 11 (as shown in Fig. 9). Details of how this value was obtained have been explained in Sects. 4.3.1 and 4.3.2. Hence, unnecessary repetitions will be avoided here. Figure 10 also shows the attainment of the optimal model parameters with

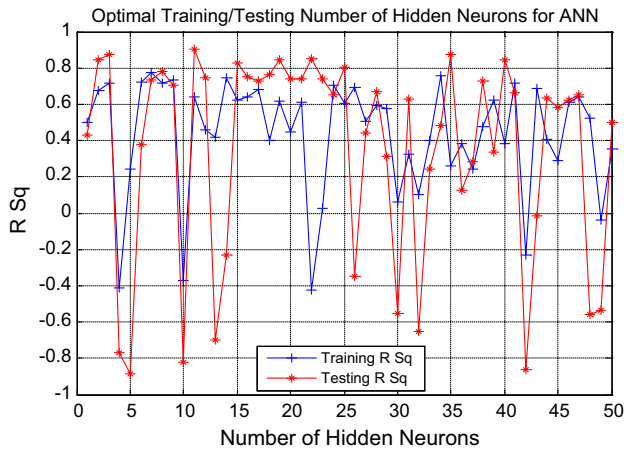


Fig. 9 Optimal number of hidden neurons for Model 2-1.

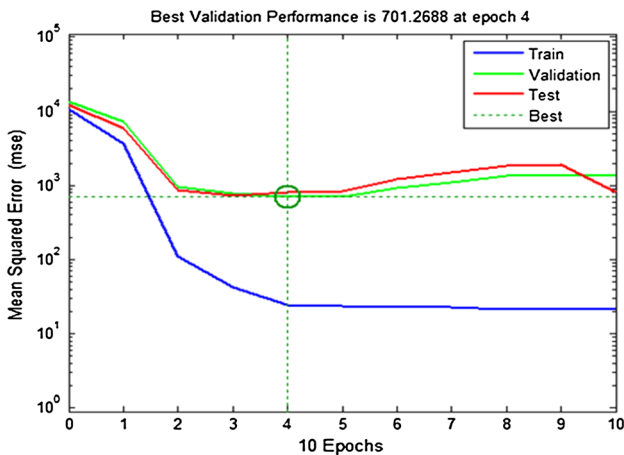


Fig. 10 Results of the training and validation for Model 2-1.

respect to the least validation error. According to the plot, since the validation error could not converge after 10 epochs, the least validation error attained at epoch 1 was chosen by the learning algorithm.

4.3.4 Prediction of M_{res} with Randomized Data Stratification

Similar to model 2-1, this model (called model 2-2) directly predicts the residual flexural strength (M_{res}) without using C_f but with the randomized stratification of the dataset. Like in model 1-2, 70 % of the data was randomly selected for training while the remaining percentage was used to test the generalization capability of the model. The randomized stratification procedure resulted in data subsets that are representative of the experimental cases. More of such details have been given in Sects. 4.3.1. and 4.3.2.

The optimal number of hidden neurons for this model was found to be 26 (as shown in Fig. 11) while the least validation error was attained at epoch 1. The model was chosen when the validation error did not improve up to 7 epochs. This is shown in Fig. 12.

After the implementation of the ANN models, the results are presented and discussed in Sect. 5.0.

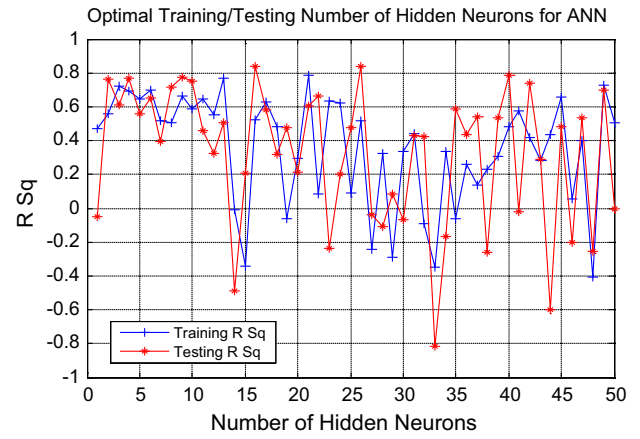


Fig. 11 Optimal number of hidden neurons for Model 2-2.

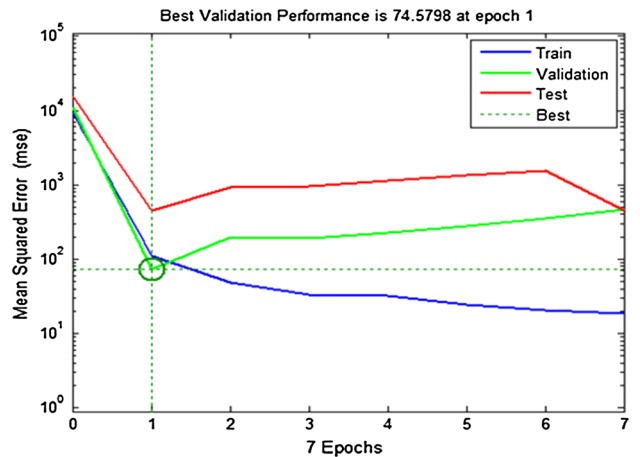


Fig. 12 Results of the training and validation for Model 2-2.

5. Results and Discussion

Since the testing process simulates the capability of the models to generalize on new and never-seen-before cases, we focus on the testing performance of the models in the presentation and analysis of the results. Based on the methodology described in Sect. 4, the results of the comparative R^2 and RMSE that were obtained from the indirect prediction of flexural strength through C_f and the direct prediction of M_{res} for the testing data subset (representing new measurements) are shown in Figs. 13–16.

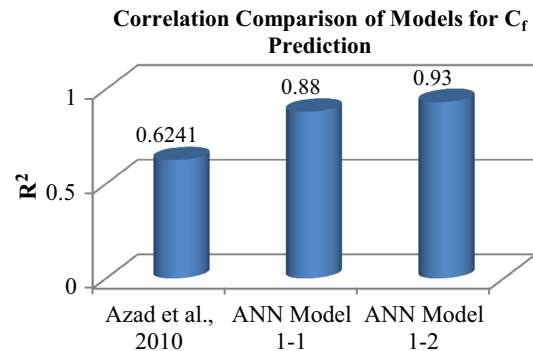


Fig. 13 R^2 Comparison of empirical model and ANN models for C_f prediction.

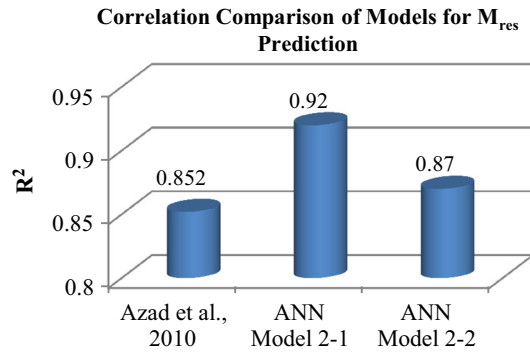


Fig. 14 R^2 of empirical model and ANN models for M_{res} prediction.

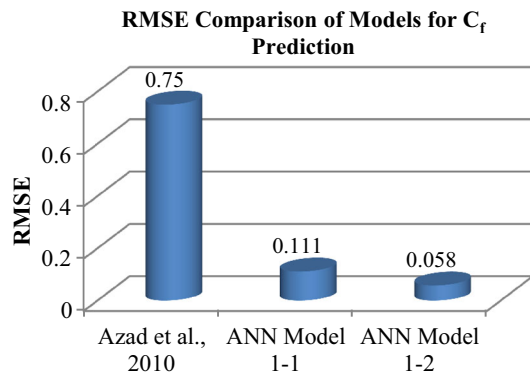


Fig. 15 RMSE Comparison of empirical model and ANN models for C_f prediction.

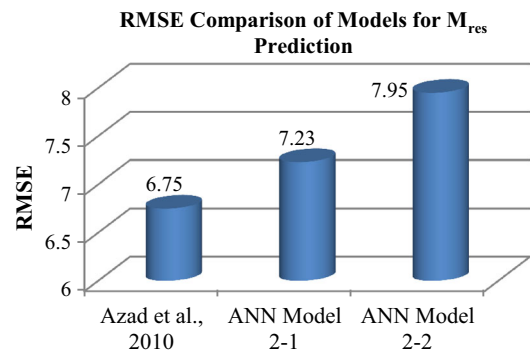


Fig. 16 RMSE Comparison of empirical model and ANN models for M_{res} prediction.

Cross-plots showing the degree of correlation between the ANN predicted results and the experimental values of C_f and M_{res} are presented in Figs. 17–20.

Figure 13 compares the R^2 of the empirical model by Azad et al. (2010) with those of the ANN models in the prediction of C_f . The figure showed that the two ANN models performed better than the Azad's empirical equation. The R^2 of model 1-1 had a 41 % improvement while Model 1-2 had a 49 % improvement over the empirical equation. This also confirmed that the ANN model with a randomized data stratification (Model 1-2) performed better than the other one with fixed stratification (Model 1-1). For the prediction of M_{res} , Fig. 14 also showed that the ANN models

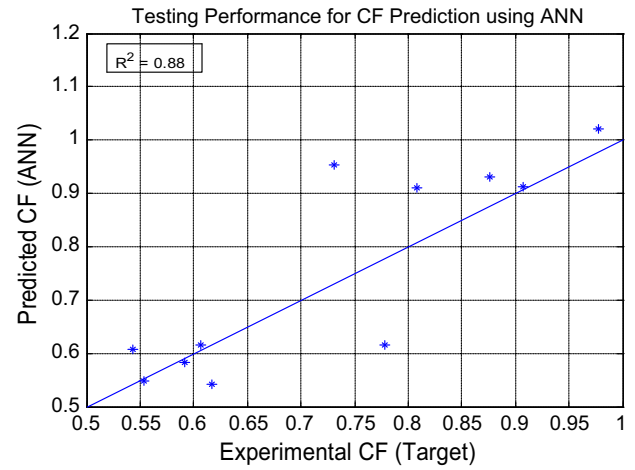


Fig. 17 Experimental and ANN prediction of M_{res} for testing data with fixed stratification.

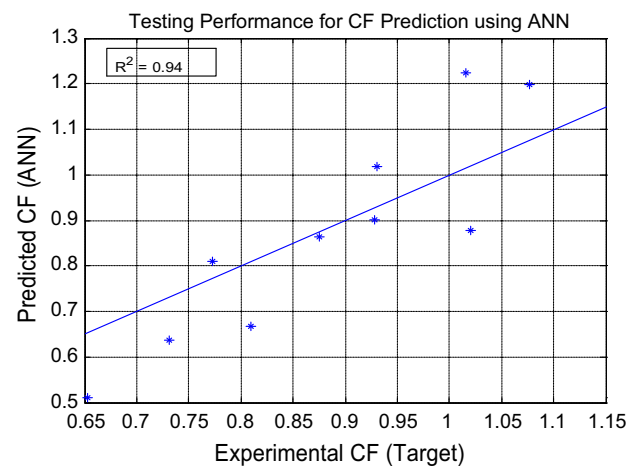


Fig. 18 Experimental and ANN prediction of M_{res} for testing data with random stratification.

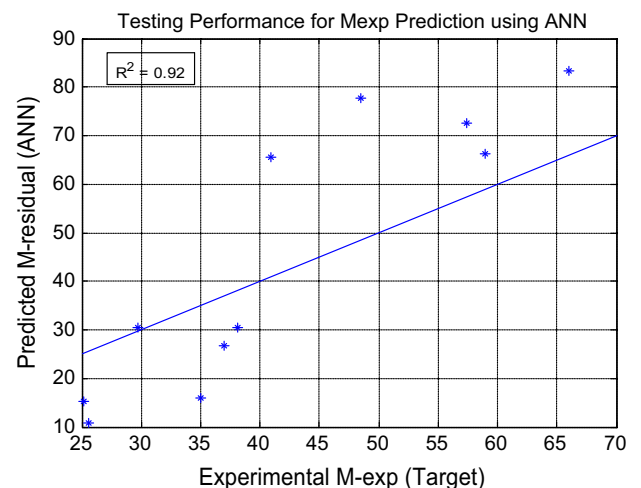


Fig. 19 Experimental and ANN prediction of M_{res} for testing data with fixed stratification.

predicted better than the Azad's equation with Model 2-1 having a 10 % improvement and Model 2-2 having a 2 % improvement over the empirical equation. However,

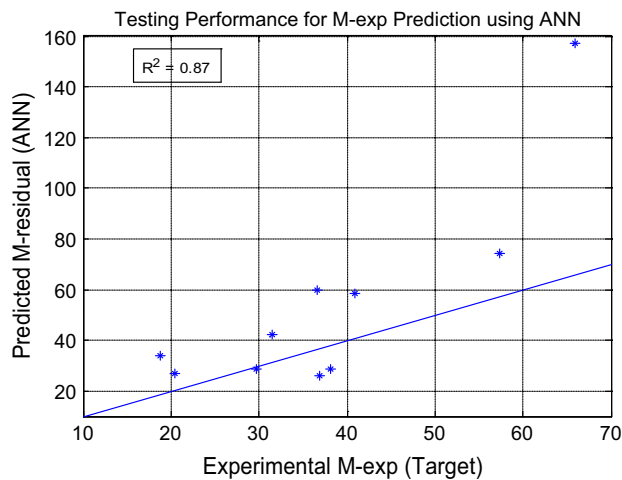


Fig. 20 Experimental and ANN prediction of M_{res} for testing data with random stratification.

between the ANN models, the one with the fixed data stratification (Model 2-1) showed better correlation with experimental data than the other one with randomized stratification (Model 2-2).

In terms of RMSE for the prediction of C_f , Fig. 15 agreed with the R^2 results by having the RMSE of the ANN models lower than that of the empirical equation. Model 2-1 and 2-2 respectively had an 85 and a 92 % reduction in error over the empirical equation. This also agrees with the R^2 results in Fig. 13 that the ANN model with randomized data stratification performed better than that with fixed stratification. For the prediction of M_{res} , Fig. 16 showed that the ANN models, despite their better performance in terms of higher R^2 , had higher errors than the empirical equation. This implies that the ANN models had higher errors associated with their better predictions.

In the overall, the comparative results showed that the ANN models that were used to predict C_f exhibited better predictive capabilities than the empirical equation. However, for optimal results, the ANN model with randomized data stratification is preferred. Also, we recommend that it is better to estimate the flexural strength of corroded RC beams indirectly through the prediction of C_f . This implies that, according to our study, going through the prediction of C_f is a better way to predict the residual flexural strength of corroded RC beams.

With the emergence of the ANN model with randomized data stratification as a better tool to predict the flexural strength of RC beams, we further analyze the degree of correlation of the prediction results of the ANN models with experimental values. Figure 17 showed that the prediction results of Model 1-1 were in better agreement with the experimental values falling in the extremes than those in the middle. C_f values that are less than 0.7 and more than 0.8 were predicted more accurately than those that fall between the two. However, Fig. 18 showed the opposite as C_f values between 0.85 and 0.95 were predicted more accurately than those in the extremes. For the prediction of M_{res} , Fig. 19 showed that despite the higher correlation, the prediction errors are high but evenly distributed over the values.

However, Fig. 20 shows that despite having the least R^2 among the ANN models, the prediction errors are low and consistent over the values except at the highest extreme. Since few samples fall in the highest extremes, the overall prediction error became low. This further confirmed that the models with random data stratification perform better than those with fixed stratification.

Giving more credence to the preference of the random data stratification over the fixed version is the visual comparison of the results of the search for the optimal number of hidden neurons for the prediction of C_f (Figs. 5 and 7) and M_{res} (Figs. 9 and 11). It would be observed that Fig. 5 (fixed stratification) is more haphazard in its fluctuation while Fig. 7 (random stratification) is smoother and more consistent. Similarly, Fig. 9 (fixed stratification) has sharper fluctuations than Fig. 11 (random stratification), which is smoother and more consistent. This behavior is probably due to the bias created by using fixed stratification instead of the fairer and unbiased randomized stratification. In the latter, there is fairness as each sample in the data has equal chance of being selected for training or testing. This perfectly agrees with the theory of randomization (Arora and Barak 2009; Cormen et al. 2001) and an existing study related to the subject (Helmy et al. 2010).

This fair analysis is based on the results obtained from the ANN models using the data obtained from a previous experimental work that was used to develop the empirical equation. Since datasets used by other authors are not available in the public domain, they could not be obtained for further testing of our ANN models. However, we hypothesize that, with the successful performance of the proposed ANN models on the available dataset, a similar successful performance is expected to be recorded on a wide array of other experimental datasets. Effort will be made in our future work to obtain such datasets for the purpose of testing the proposed models.

6. Conclusions, Limitation and Future Work

In order to reduce the cost, effort and time associated with persistent laboratory experiments, with the aim of benefiting from the high volume of experimental data gathered over the years and to increase the degree of accuracy of predictive models, we presented four optimized ANN models to predict the residual flexure strength of corroded RC beams with the diameter of reinforced steel (D) and corrosion activity index ($I_{corr}T$) as input variables. Computational techniques, especially ANN, have the capability to handle the non-linear relationship between predictor variables and their target values. A number of empirical equations have been proposed in literature. However, since they are based on assumed linear relationships among various predictor variables, they could not adequately handle such a natural phenomenon as corrosion.

This study was conducted in two parts: direct prediction of the residual strength, M_{res} and indirect prediction through the correction factor, C_f . Each part was further experimented with fixed and randomized stratification of the data into training

and testing subsets. To ensure fairness in comparison and to increase the confidence in the outcome of the study, the same data that was used by Azad et al. (2010) was used to train and evaluate the performance of the ANN models. From the rigorous analysis of the ANN results and a comparison of the results with those of Azad's empirical equation, the ANN models demonstrated superior performance.

The outcome of this study can be highlighted as follows:

- With their higher coefficients of determination, the ANN models could be alternative modeling tools to the empirical equation of Azad et al. (2010) in the prediction of flexural strength of corroded RC beams. This is due to the excellent learning capability and the dynamic nature of ANN. Empirical correlations are static and have no learning capability.
- The ANN models with randomized data stratification gave better predictions than those with fixed stratification. This is due to the fairness of the randomization process.
- The ANN models with C_f as the target variable gave higher prediction accuracies and reduced errors than those that directly predict M_{res} . Hence, we recommend to Construction Engineers to estimate the residual flexural strength of RC beams by multiplying the correction factor, C_f , predicted by ANN, with the theoretical moment capacity, $M_{th,c}$.
- Our study demonstrated that ANN models are simpler, adaptive and more reliable tools for the prediction of flexural strength of corroded RC beams.

Since the main objective of this work is to investigate the capability of ANN models to predict the flexural strength of reinforced concrete, the focus is to keep the algorithm design and implementation simple. This follows the Occam's Razor principle (Jefferys and Berger 1991). Investigating the effect of various parameters (such as the effect of data normalization, activation function, different learning algorithms and different layers) on the performance of the ANN models will be carried out in our future work. More complex learning paradigms such as hybrid and ensemble concepts will also be considered.

The authors plan to confirm the consistency of the results of this study in the continued and future work by using more experimental data from various types of concrete samples and published datasets from previous studies. In the continued search for better predictive tools, it is planned to implement other types of ANN such as Generalized Regression Neural Networks, Radial Basis Functional Networks and Functional Networks. More advanced computational intelligence techniques such as Extreme Learning Machines, Support Vector Machine and Type-2 Fuzzy Logic Systems will also be considered.

Acknowledgement

The authors would like to acknowledge the support of King Fahd University of Petroleum and Minerals (KFUPM) for providing the resources used in the conduct of this research.

Open Access

This article is distributed under the terms of the Creative Commons Attribution License which permits any use, distribution, and reproduction in any medium, provided the original author(s) and the source are credited.

References

- Abdalla, J. A., Elsanosi, A., & Abdelwahab, A. (2007). Modeling and simulation of shear resistance of R/C beams using artificial neural network. *Journal of the Franklin Institute*, 344(5), 741–756.
- Al-Gohi, B. H. A. (2008). *Time-dependent modeling of loss of flexural strength of corroding RC beams*. Master Thesis, King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia.
- Arora, S., & Barak, B. (2009). *Computational complexity: a modern approach* (1st ed.). Cambridge, UK: Cambridge University Press.
- Azad, A., Ahmad, S., & Al-Gohi, B. (2010). Flexural strength of corroded reinforced concrete beams. *Magazine of Concrete Research*, 62(6), 405–414.
- Azad, A., Ahmad, S., & Azher, S. A. (2007). Residual strength of corrosion-damaged reinforced concrete beams. *ACI Material Journal*, 104(1), 40–47.
- Baughman, D. R. (1995). *Neural networks in bioprocessing and chemical engineering*. PhD Dissertation, Virginia Tech, Blacksburg, VA.
- Beale, M., & Demuth, H. (2013). *Neural network toolbox user's guide*. Natick, MA: The Mathworks Inc.
- Bies, R. R., Muldoon, M. F., Pollock, B. G., Manuck, S., Smith, G., & Sale, M. E. (2006). A genetic algorithm-based hybrid machine learning approach to model selection. *Journal of Pharmacokinetics and Pharmacodynamics*, 33(2), 195–221.
- Cabrera, J. (1996). Deterioration of concrete due to reinforcement steel corrosion. *Cement & Concrete Composites*, 18(1), 47–59.
- Castillo, E., Gutiérrez, J. M., Hadi, A. S., & Lacruz, B. (2001). Some applications of functional networks in statistics and engineering. *Technometrics*, 43, 10–24.
- Chen, H., Tsai, K., Qi, G., Yang, J., & Amini, F. (1995). Neural network for structure control. *Journal of Computing in Civil Engineering*, 9(2), 168–176.
- Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2001). *Introduction to algorithms*. Cambridge, MA: MIT press.
- Coronelli, D., & Gambarova, P. (2004). Structural assessment of corroded reinforced concrete beams: modeling guidelines. *Journal of Structural Engineering*, 130(8), 1214–1224.
- Eskandari, H., Rezaee, M. R., & Mohammadnia, M. (2004). Application of multiple regression and artificial neural network techniques to predict shear wave velocity from wireline log data for carbonate reservoir, South-West Iran. *CSEG Recorder*, 42, 48.

- Flood, I., & Kartam, N. (1994). Neural networks in civil engineering. II: Systems and application. *Journal of Computing in Civil Engineering*, 8(2), 149–162.
- Guler, I. (2005). ECG beat classifier designed by combined neural network model. *Pattern Recognition*, 38(2), 199–208.
- Hasancebi, O., & Dumlupinar, T. (2013). Linear and nonlinear model updating of reinforced concrete T-beam bridges using artificial neural networks. *Computers & Structures*, 119, 1–11.
- Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning: data mining, inference, and prediction* (2nd ed.). Berlin, Germany: Springer.
- Helmy, T., Anifowose, F. A., & Sallam, E. S. (2010). An efficient randomized algorithm for real-time process scheduling in PicOS operating system. In K. Elleithy (Ed.), *Advanced techniques in computing sciences and software engineering* (pp. 117–122). New York, NY: Springer.
- Hsu, D. S., & Chung, H. T. (2002). Diagnosis of reinforced concrete structural damage base on displacement time history using the back-propagation neural network technique. *Journal of Computing in civil engineering*, 16(1), 49–58.
- Huang, R., & Yang, C. (1997). Condition assessment of reinforced concrete beams relative to reinforcement corrosion. *Cement & Concrete Composites*, 19(2), 131–137.
- Inan, O. T., Giovangrandi, L., & Kovacs, G. T. (2006). Robust neural-network-based classification of premature ventricular contractions using wavelet transform and timing interval features. *IEEE Transactions on Biomedical Engineering*, 53(12), 2507–2515.
- Inel, M. (2007). Modeling ultimate deformation capacity of RC columns using artificial neural networks. *Engineering Structures*, 29(3), 329–335.
- Jefferys, W. H. & Berger, J. O. (1991). *Sharpening Ockham's Razor on a Bayesian Strop*. Technical Report #91-44C, Department of Statistics, Purdue University, West Lafayette, IN.
- Jin, W.-L., & Zhao, Y.-X. (2001). Effect of corrosion on bond behavior and bending strength of reinforced concrete beams. *Journal of Zhejiang University (Science)*, 2(3), 298–308.
- Kang, H. T., & Yoon, C. J. (1994). Neural network approaches to aid simple truss design problems. *Computer-Aided Civil and Infrastructure Engineering*, 9(3), 211–218.
- Kirkegaard, P. H. & Rytter, A. (1994). Use of neural networks for damage assessment in a steel mast. In *Proceedings of the 12th International Modal Analysis Conference of the Society for Experimental Mechanics*. Honolulu, HI.
- Li, L., & Jiao, L. (2002). Prediction of the oilfield output under the effects of nonlinear factors by artificial neural network. *Journal of Xi'an Petroleum Institute*, 17(4), 42–44.
- Mangat, P. S., & Elgarf, M. S. (1999). Flexural strength of concrete beams with corroding reinforcement. *ACI Structural Journal*, 96(1), 149–158.
- Moghadassi, A., Parvizian, F., Hosseini, S. M., & Fazlali, A. (2009). A new approach for estimation of PVT properties of pure gases based on artificial neural network model. *Brazilian Journal of Chemical Engineering*, 26(1), 199–206.
- Mohaghegh, S. (1995). Neural network: What it can do for petroleum engineers. *Journal of Petroleum Technology*, 47(1), 42–42.
- Nascimento, C. A. O., Giudici, R., & Guardani, R. (2000). Neural network based approach for optimization of industrial chemical processes. *Computers & Chemical Engineering*, 24(9), 2303–2314.
- Neaupane, K. M., & Adhikari, N. (2006). Prediction of tunneling-induced ground movement with the multi-layer perceptron. *Tunnelling and Underground Space Technology*, 21(2), 151–159.
- Nokhasteh, M. A., & Eyre, J. R. (1992) The effect of reinforcement corrosion on the strength of reinforced concrete members. In *Proceedings of Structural integrity assessment*. London, UK: Elsevier Applied Science.
- Ou, Y. C., Tsai, L. L., & Chen, H. H. (2012). Cyclic performance of large-scale corroded reinforced concrete beams. *Earthquake Engineering and Structural Dynamics*, 41(4), 593–604.
- Pandey, P., & Barai, S. (1995). Multilayer perceptron in damage detection of bridge structures. *Computers & Structures*, 54(4), 597–608.
- Petrus, J. B., Thuijsman, F., & Weijters, A. J. (1995). *Artificial neural networks: An introduction to ANN theory and practice*. Berlin, Germany: Springer.
- Phung, S. L., & Bouzerdoum, A. (2007). A pyramidal neural network for visual pattern recognition. *IEEE Transactions on Neural Networks*, 18(2), 329–343.
- Rafiq, M., Bugmann, G., & Easterbrook, D. (2001). Neural network design for engineering applications. *Computers & Structures*, 79(17), 1541–1552.
- Ravindrarajah, R. S., & Ong, K. (1987). Corrosion of steel in concrete in relation to bar diameter and cover thickness. *ACI Special Publication*, 100, 1667–1678.
- Revathy, J., Suguna, K., & Raghunath, P. N. (2009). Effect of corrosion damage on the ductility performance of concrete columns. *American Journal of Engineering and Applied Sciences*, 2(2), 324–327.
- Rodriguez, J., Ortega, L., & Casal, J. (1997). Load carrying capacity of concrete structures with corroded reinforcement. *Construction and Building Materials*, 11(4), 239–248.
- Tachibana, Y., Maeda, K.-I., Kajikawa, Y., & Kawamura, M. (1990). Mechanical behavior of RC beams damaged by corrosion of reinforcement. *Elsevier Applied Science*, 178–187.
- Tsai, C.-H., & Hsu, D.-S. (2002). Diagnosis of reinforced concrete structural damage base on displacement time history using the back-propagation neural network technique. *Journal of Computing in Civil Engineering*, 16(1), 49–58.
- Übeyli, E. D. (2009). Combined neural network model employing wavelet coefficients for EEG signals classification. *Digital Signal Processing*, 19(2), 297–308.
- Uomoto, T., & Misra, S. (1988). Behavior of concrete beams and columns in marine environment when corrosion of reinforcing bars takes place. *ACI Special Publication*, 109, 127–146.

- VanLuchene, R., & Sun, R. (1990). Neural networks in structural engineering. *Computer-Aided Civil and Infrastructure Engineering*, 5(3), 207–215.
- Wang, X. H., & Liu, X. L. (2008). Modeling the flexural carrying capacity of corroded RC beam. *Journal of Shanghai Jiaotong University (Science)*, 13(2), 129–135.
- Waszczyszyn, Z., & Ziemiański, L. (2001). Neural networks in mechanics of structures and materials—new results and prospects of applications. *Computers & Structures*, 79(22), 2261–2276.
- Wolpert, D. H., & Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1), 67–82.
- Wu, X., Ghaboussi, J., & Garrett, J. H. (1992). Use of neural networks in detection of structural damage. *Computers & Structures*, 42(4), 649–659.